```
#include <stdio.h>
int main()
{
    printf("Hola");
    return 0;
}
```

```
#include <stdio.h>

int main()
{
    printf("Hola");
    return 0;
}
```

Directivas del Preprocesador:

- No son parte del código ejecutable
- Son "indicaciones" que se dan al proceso de compilación
- Las directivas #include importan la "documentación" de las funciones a utilizar, listadas en "archivos de cabecera" *.h
- También se utilizarán mucho las directivas #define

```
#include <stdio.h>
```

```
int main()
{
    printf("Hola");
    return 0;
}
```

Función main() -> Programa principal

- Debe existir en cualquier programa ejecutable
- Consta del encabezado int main()
- y del código ejecutable encerrado entre llaves {...}
- El código se ejecutará ordenadamente desde la llave superior { (inicio del algoritmo) hasta la llave inferior } (fin del algoritmo)

```
#include <stdio.h>

int main()
{
    printf("Hola");
    return 0;
}
```

```
#include <stdio.h>

int main()
{
    printf("Hola");
    return 0;
}
```

código ejecutable

- Las sentencias terminan en ;
- Contienen expresiones, operaciones y llamadas a funciones
- En un código típico también habrán:
 - estructuras para el control de flujo
 - Declaración de variables

```
#include <stdio.h>
Llamada a la función printf()

La función printf() equivale a la instrucción Escribir del pseudocódigo

Solo puede utilizarse si se incluye stdio.h

return 0;
}
```

```
#include <stdio.h>
int main()
{
    printf("Hola");
    return 0;
}
```

Instrucción de retorno

- Su ejecución finaliza con el programa
- Debe ir al final del programa principal (antes de la llave de cierre })

Nivel	Operadores	Descripción	Asoci.
1	() [] -> .	Acceso a un elemento de un vector y paréntesis	Izquierdas
2	+ - ! ~ * & ++ (cast) sizeof	Signo (unario), negación lógica, negación bit a bit Acceso a un elemento (unarios): puntero y dirección Incremento y decremento (pre y post) Conversión de tipo (<i>casting</i>) y tamaño de un elemento	Derechas
3	* / %	Producto, división, módulo (resto)	Izquierdas
4	+ -	Suma y resta	Izquierdas
5	>> <<	Desplazamientos	Izquierdas
6	< <= >= >	Comparaciones de superioridad e inferioridad	Izquierdas
7	== !=	Comparaciones de igualdad	Izquierdas
8	٤	Y (And) bit a bit (binario)	Izquierdas
9	*	O-exclusivo (<i>Exclusive-Or</i>) (binario)	Izquierdas
10		O (Or) bit a bit (binario)	Izquierdas
11	6.6	Y (And) lógico	Izquierdas
12	П	O (Or) lógico	Izquierdas
13	?:	Condicional	Derechas
14	= *= /= %= += -= >>= <<= &= ^= =	Asignaciones	Derechas
15	,	Coma	Izquierdas

Operadores de C

Nivel	Operadores	Descripción	Asoci.
1	() [] -> .	Acceso a un elemento de un vector y paréntesis	Izquierdas
2	+ - ! ~ * & ++ (cast) sizeof	Signo (unario), negación lógica, negación bit a bit Acceso a un elemento (unarios): puntero y dirección Incremento y decremento (pre y post) Conversión de tipo (<i>casting</i>) y tamaño de un elemento	Derechas
3	* / %	Producto, división, módulo (resto)	Izquierdas
4	+ -	Suma y resta	Izquierdas
5	>> <<	Desplazamientos	Izquierdas
6	< <= >= >	Comparaciones de superioridad e inferioridad	Izquierdas
7	== !=	Comparaciones de igualdad	Izquierdas
8	&	Y (And) bit a bit (binario)	Izquierdas
9	*	O-exclusivo (<i>Exclusive-Or</i>) (binario)	Izquierdas
10		O (Or) bit a bit (binario)	Izquierdas
11	66	Y (And) lógico	Izquierdas
12	П	O (Or) lógico	Izquierdas
13	2:	Condicional	Derechas
14	= *= /= %= += -= >>= <<= &= ^= =	Asignaciones	Derechas
15	,	Coma	Izquierdas

Operadores Unarios (un único operando)

Operadores Aritméticos

Operadores relacionales

Operadores lógicos (AND/OR)

Operadores de asignación

```
#include <stdio.h>
int main()
    /* Declaración de variables */
    int Npizzas;
    float harina, sal, aceite, levadura, azucar, agua;
    /* Algoritmo */
    printf("Ingrese la cantidad de pizzas: ");
    scanf("%d", &Npizzas);
    harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

```
azucar = Npizzas/5.0;
agua = Npizzas/5.0 * 700;
printf("Para amasar %d pizzas necesita:\n", Npizzas);
printf("%f Kg de harina 000\n", harina);
printf("%f cucharadas de sal\n", sal);
printf("%f cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

```
#include <stdio.h>
int main()
    /* Declaración de variables *,
    int Npizzas;
    float harina, sal, aceite, levadura, azuc
    /* Algoritmo */
    printf("Ingrese la cantidad de pizzas: ");
    scanf("%d", &Npizzas);
    harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

Comentarios

Es tratado como texto y no se compila

encerrados entre /* y */

azud

agua

prin

Pueden tener varias lineas

```
printf("%f cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

```
#include <stdio.h>
int main()
    /* Declaración de variables */
    int Npizzas;
    float harina, sal, aceite, levadura, azucar, agua;
    /* Algoritmo */
    printf("Ingrese la cantidad de pizzas: ");
    scanf("%d", &Npizzas);
    harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

Declaración de variables

- Se deben declarar **TODAS** las variables antes de cualquier sentencia ejecutable
- El formato es:
 - <modificadores de tipo> <tipo> <nombre_variable>
- El identificador de la variable puede contener letras, números y guiones bajos (_)
- Solo puede comenzar con letras o guión bajo

Tipos de variables

Los tipos básicos son;

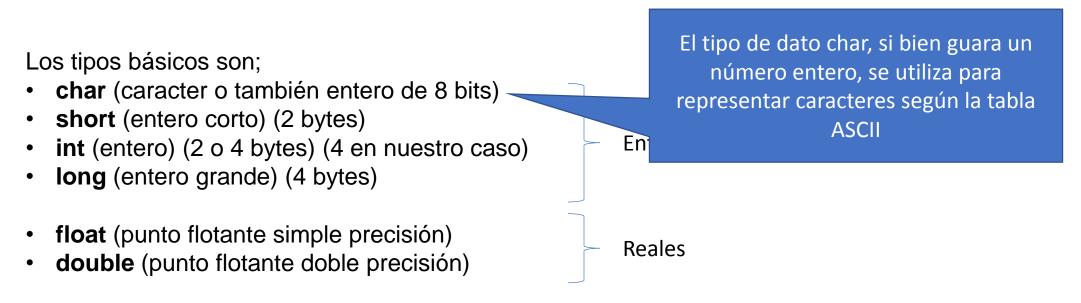
- char (caracter o también entero de 8 bits)
- short (entero corto) (2 bytes)
- int (entero) (2 o 4 bytes) (4 en nuestro caso)
- long (entero grande) (4 bytes)
- float (punto flotante simple precisión)
- double (punto flotante doble precisión)

Enteros

Reales

Podemos usar el modificador **unsigned** en la declaración de char, short, int y long, para declararlo como variable entera **sin signo**

Tipos de variables



Podemos usar el modificador **unsigned** en la declaración de char, short, int y long, para declararlo como variable entera **sin signo**

Tipos de variables

char	-128 to 127
unsigned char	0 to 255
short int	-32,768 to 32,767
unsigned short int	0 to 65,535
int	-2,147,483,648 to 2,147,483,647
unsigned int	0 to 4,294,967,295
long int	-2,147,483,648 to 2,147,483,647
unsigned long int	0 to 4,294,967,295
float	1.17×10 ⁻³⁸ to 3.40×10 ³⁸
double	2.22×10 ⁻³⁰⁸ to 1.79×10 ³⁰⁸

(Caracteres ASCII Caracteres ASCII								AS	CII e	xtendi	do					
	de	control		i	mpri	mible	S					(Págir	na de	códig	10 437	7)	
00	NULL	(carácter nulo)	32	espacio	64	@	96	•	1	128	Ç	160	á	192	L	224	Ó
01	SOH	(inicio encabezado)	33	!	65	A	97	a	П	129	ű	161	i	193	1	225	ß
02	STX	(inicio texto)	34		66	В	98	b	П	130	é	162	ó	194	т	226	Ô
03	ETX	(fin de texto)	35	#	67	C	99	С	П	131	â	163	ú	195	ŀ	227	Ò
04	EOT	(fin transmisión)	36	\$	68	D	100	d	П	132	ä	164	ñ	196	-	228	ő
05	ENQ	(consulta)	37	%	69	E	101	е	П	133	à	165	Ñ	197	+	229	Õ
06	ACK	(reconocimiento)	38	&	70	F	102	f		134	à	166	a	198	ã	230	μ
07	BEL	(timbre)	39		71	G	103	g		135	ç	167	0	199	Ã	231	þ
08	BS	(retroceso)	40	(72	Н	104	h	П	136	ê	168	ė	200	L	232	Þ
09	HT	(tab horizontal)	41)	73	1	105	i		137	ë	169	®	201	1	233	Ú
10	LF	(nueva línea)	42	*	74	J	106	j		138	è	170	7	202	ᅶ	234	Û
11	VT	(tab vertical)	43	+	75	K	107	k		139	ï	171	1/2	203	TE	235	Ù
12	FF	(nueva página)	44		76	L	108	- 1		140	î	172	1/4	204	-	236	Ý
13	CR	(retorno de carro)	45	7.53	77	M	109	m		141	ì	173	i	205	-	237	Ý
14	SO	(desplaza afuera)	46		78	N	110	n	П	142	Ä	174	44	206	#	238	
15	SI	(desplaza adentro)	47	- 1	79	0	111	0	П	143	A	175	» —	207	Ħ	239	**
16	DLE	(esc.vínculo datos)	48	0	80	Р	112	p		144	É	176	200	208	ð	240	=
17	DC1	(control disp. 1)	49	1	81	Q	113	q	п	145	æ	177	artists.	209	Đ	241	±
18	DC2	(control disp. 2)	50		82	R	114	r	П	146	Æ	178		210	Ê	242	_
19	DC3	(control disp. 3)	51	3	83	S	115	S		147	ô	179		211	Ë	243	3/4
20	DC4	(control disp. 4)	52		84	Т	116	t		148	Ö	180	+	212	È	244	9
21	NAK	(conf. negativa)	53		85	U	117	u		149	Ò	181	A	213	1	245	9
22	SYN	(inactividad sínc)	54		86	V	118	٧		150	û	182	A	214	1	246	÷
23	ETB	(fin bloque trans)	55	7	87	W	119	W	П	151	ù	183	A	215	Ī	247	3
24	CAN	(cancelar)	56		88	X	120	X	ш	152	ÿ	184	©	216	1	248	0
25	EM	(fin del medio)	57	9	89	Y	121	у	П	153	O	185	4	217	1	249	65%
26	SUB	(sustitución)	58		90	Z	122	Z	н	154	Ü	186		218	Г	250	
27	ESC	(escape)	59		91	[123	{	п	155	Ø	187	7	219		251	1
28	FS	(sep. archivos)	60		92	1	124	1	ш	156	£	188	1	220		252	3
29	GS	(sep. grupos)	61		93]	125	}		157	Ø	189	¢	221	1	253	2
30	RS	(sep. registros)	62		94	٨	126	~		158	×	190	¥	222	1	254	
31	US	(sep. unidades)	63	?	95	_				159	f	191	٦	223		255	nbsp
127	DEL	(suprimir)															

32 espacio 64 @ 96 128 C 160 6 192 L 224 0 0 0 1 SOH (inicio encabezado) 33 65 A 97 a 129 u 161 i 193 L 225 B 0 0 0 1 1 1 1 1 2 2 0 0 0 0 0 0 0 0	(eres ASCII control		Caracteres ASCII imprimibles										ASCII extendido (Página de código 437)						
O1 SOH (inicio encabezado) O2 STX (inicio texto) O3 ETX (fin de texto) O4 O5 C O5	0.0			0.0							400				-	10 40	_	Á	ı		
02 STX	0.000		\$300 WYS0 WYS0 WYS0 WYS	5/9/27			11000	427972				15	100000000000000000000000000000000000000	13700	100000000000000000000000000000000000000	-	974 055454		ı		
03 ETX (fin de texto) 04 EOT (fin transmision) 05 ENQ (consuta) 06 ACK (reconocimiento) 07 BEL (timbre) 08 BS (retroceso) 09 HT (tab horizontal) 10 LF (nueva linea) 11 VT (tab vertical) 12 FF (nueva página) 13 CR (retorno de carro) 14 SO (desplaza adentro) 15 SI (desplaza adentro) 16 DLE (esc. vínculo datos) 17 DC1 (control disp. 2) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. rgripos) 30 RS (sep. rgripos) 31 US (sep. nuddeds) 60 2 99 C 1311 â 165 û 195 227 Ò 00 d 132 â 165 û 196 196 197 228 0 0100 d 132 â 165 û 196 196 197 229 0 02 d 132 â 165 û 197 229 0 03 18 & 70 F 100 10 d 132 â 165 û 196 196 197 229 0 04 12 13 â 165 û 197 229 0 04 14 1 17 14 14 14 14 14 14 14 14 14 14 14 14 14	100000000000000000000000000000000000000		CONTRACTOR OF THE PROPERTY OF	100	15.555	and the latest terminal		and the same of th				1	and the same of th		and the second section		and the state of the state of		П		
04 EOT (fin transmisión) 05 ENQ (consulta) 06 ACK (reconocimiento) 07 BEL (timbre) 08 BS (retroceso) 09 HT (tab horizontal) 10 LF (nueva línea) 11 VT (tab vertical) 12 FF (nueva página) 13 CR (retromo de carro) 14 SO (desplaza adentro) 15 SI (desplaza adentro) 16 DLE (esc.vinculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 25 SIM (fin loque trans) 25 SIM (inactividad sinc) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 60 S S 68 D 100 d 132 ä 164 ñ 196 - 228 ö 101 e 133 à 165 Ñ 197 + 229 Ö 10 Caracteres numéricos Desde el carácter '0' al carácter '9' ordenados en forma creciente 45 - 77 M 46 . 78 47 / 79 48 0 80 P 112 P 144 E 176 208 0 240 = 117 11 E 243 ½ 12 P 144 E 176 209 D 241 ± 11 ± 11 ± 11 ± 11 ± 11 ± 11 ± 11	0.000000			2557				100000000000000000000000000000000000000					95/20/20/20	200	1075 A Q Q D S		0.0000000000000000000000000000000000000		L		
05 ENQ (consulta) 06 ACK (reconocimiento) 07 BEL (timbre) 08 BS (retroceso) 09 HT (tab horizontal) 10 LF (nueva inea) 11 VT (tab vertical) 12 FF (nueva página) 13 CR (retorno de carro) 14 SO (desplaza adentro) 16 DLE (esc. vinculo datos) 17 DC1 (control disp. 2) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. grigistros) 37 % 69 E 101 e 133 à 165 Ñ 197	and the second second		Control of the Contro	and bearing		and the last time	10.00	and the second	- 77				and the second	11/20	and the state of t	F		_	ı		
06 ACK (reconocimiento) 07 BEL (timbre) 08 BS (retroceso) 09 HT (tab horizontal) 10 LF (nueva página) 11 VT (tab vertical) 12 FF (nueva página) 13 CR (retorno de carro) 14 SO (desplaza adeutro) 16 DLE (esc.vinculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 3) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. grajstros) 40 (72 H 10 (72 H 10 (72 H 10)	0.0000		The state of the s	12700	1207		100	100 (000)				1771	- Little Co.	154	100000000000000000000000000000000000000				П		
07 BEL (timbre) 08 BS (retroceso) 09 HT (tab horizontal) 10 LF (nueva linea) 11 VT (tab vertical) 12 FF (nueva página) 13 CR (retorno de carro) 14 SO (desplaza adentro) 15 SI (desplaza adentro) 16 DLE (esc. vinculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 3) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 29 GS (sep. grupos) 30 RS (sep. registros) 60 < 92 V 124 I 156 £ 188			AND THE RESERVE OF THE PROPERTY OF THE PROPERT		1000	137 57		100000000000000000000000000000000000000	е			a	- COCC	N	The second secon	+	and the second second	0	П		
08 BS (retroceso) 09 HT (tab horizontal) 10 LF (nueva linea) 11 VT (tab vertical) 12 FF (nueva página) 13 CR (retorno de carro) 14 SO (desplaza adentro) 15 SI (desplaza adentro) 16 DLE (esc.vinculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 25 ETB (fin bloque trans) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 40 (72 H 1 10	1000000		TO SECURIT OF STREET STREET, S	1.027.0	1,147.0			100			474		466	-	400		000				
10	100000		(timbre)				11.00														
10 LF (nueva linea) 11 VT (tab vertical) 12 FF (nueva página) 13 CR (retorno de carro) 14 SO (desplaza aduera) 15 SI (desplaza adentro) 16 DLE (esc.vínculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (find el medio) 26 SUB (sustfución) 57 9 89 Y 121 y 153 Ö 185 ⅓ 210 ■ 248 ° 255 nbsp 18 DC2 (escape) 59 ; 91 [123 { 155 Ø 187 ☐ 219 ■ 221 ■ 255 nbsp 19 DC3 (sep. grupos) 10 LF (nueva línea) 42 * 74 J 10 10 10 10 10 10 10 10 10 10 10 10 10	10000		(retroceso)	1777	7		Н	199				Cara	ctoro	s but	máric	26					
11 VT (tab vertical) 12 FF (nueva página) 13 CR (retorno de carro) 14 SO (desplaza afuera) 15 SI (desplaza adentro) 16 DLE (esc. vínculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 2) 19 DC3 (control disp. 4) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sínc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 IN RS (sep. archivos) 29 GS (sep. grupos) 31 US (sep. unidades) 43 + 75 K 10 44 , 76 L 44 , 76 L 45 - 77 M 46 . 78 47 / 79	09		(tab horizontal)			100	1	10				Cara	cteres	s IIUI	Herica	US .					
12	10		(nueva línea)					10													
13 CR (retorno de carro) 14 SO (desplaza afuera) 15 SI (desplaza adentro) 16 DLE (esc.vinculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 50 2 89 R 114 21 P 144 E 176 209 D 241 ± 17	11	VT	(tab vertical)	43	+	75	K	10					(0)			(0)					
14 SO (desplaza afuera) 15 SI (desplaza adentro) 16 DLE (esc.vínculo datos) 17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sínc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 60 < 92 \ 122 \ 124 \ 156 € 188 \ 220	12	FF	(nueva página)	44		76	L	3	Des	sde	el ca	aract	:er ' U '	al ca	aracte	r '9'	orde	nados			
14	13	CR	(retorno de carro)	45	2. 1 2	77	M		an i	for	ma c	racia	anta								
16	14	SO	(desplaza afuera)	46		78			CIII	101	IIIa C	recie	FILE								
17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 30 RS (sep. grupos) 31 US (sep. unidades) 49 1 81 Q 113 q 145 æ 177 209 € D 241 ± 145 æ 177 209 € D 241 ± 146 Æ 178 209 € D 241 ± 146 Æ 178 210 £ 242 146 Æ 178 210 £ 242 147 ô 179	15	SI	(desplaza adentro)	47	- 1	79	U	11													
17 DC1 (control disp. 1) 18 DC2 (control disp. 2) 19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 49 1 81 Q 113 q 145 æ 177 209 Ð 241 ± 145 æ 177 209 Ð 241 ± 146 Æ 178 210 Ē 242 147 ô 179	16	DLE	(esc.vínculo datos)	48	0	80	P	112	р		144	E	170	200	200	0	240	=			
19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad síne) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 51 3 83 S 115 s 147 ô 179	17	DC1	(control disp. 1)	49	1	81	Q	113	q		145	æ	177	200	209	Đ	241	±	П		
19 DC3 (control disp. 3) 20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad síne) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 51 3 83 S 115 s 147 ô 179	18	DC2	(control disp. 2)	50	2	82	R	114	7.7		146	Æ	178		210	Ê	242		П		
20 DC4 (control disp. 4) 21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 52 4 84 T 116 t 148 ö 180 - 148 i 212 È 244 ¶ 149 ò 181 À 213 i 245 § 149 ò 181 À 213 i 245 § 149 ò 181 À 213 i 245 § 149 ò 181 À 213 i 246 ÷ 148 ö 180 - 149 ò 181 À 213 i 246 ÷ 148 ö 180 - 149 ò 181 À 213 i 246 ÷ 148 ö 180 - 149 ò 181 À 213 i 246 ÷ 148 ö 180 - 149 ò 181 À 214 i 246 ÷ 148 ö 180 - 149 ò 181 À 214 i 246 ÷ 150 û 182 À 214 i 246 ÷ 151 û 183 À 215 i 247 , 152 ÿ 184 © 216 i 248 ° 153 Ö 185 - 154 Ü 186 218	19	DC3	(control disp. 3)	51	3	83	S	115	S	ш	147	ô	179	T	211	Ë	243	3/4	П		
21 NAK (conf. negativa) 22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 53 5 85 U 117 u 149 ò 181 À 213 i 245 § 4	20	DC4	127 Carlot 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	52	4	84	Т	116	t		148	Ö	180	4	212	È	244	¶	П		
22 SYN (inactividad sinc) 23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 54 6 86 V 118 v 150 û 182 Â 214 Î 246 ÷ 151 ù 183 À 215 Î 247 , 152 ÿ 184 © 216 Ï 248 ° 153 Ö 185 ╣ 217	21	NAK	A STATE OF THE PROPERTY OF THE PARTY OF THE	53	5	85	U	117	u		149	ò	181	Á	213	1	245		L		
23 ETB (fin bloque trans) 24 CAN (cancelar) 25 EM (fin del medio) 26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 55 7 87 W 119 W 151 ù 183 À 215 î 247 152 ÿ 184 © 216 ï 248 ° 153 Ö 185 ╣ 217	22	SYN	Unique la	54	6	86	V	118	V		150	û	182	Â	214	ĺ	246		П		
24 CAN (cancelar) 56 8 88 X 120 x 152 ÿ 184 © 216 ï 248 ° 25 EM (fin del medio) 57 9 89 Y 121 y 153 Ö 185 ¾ 217 J 249 " 26 SUB (sustitución) 58 : 90 Z 122 z 154 Ü 186 ¾ 217 J 249 " 27 ESC (escape) 59 ; 91 [123 { 155 Ø 187 ¾ 219 250 . 28 FS (sep. archivos) 60 <	100000		Company of the Compan	55	7	87	W	100000000000000000000000000000000000000	w			ù	100000000000000000000000000000000000000	4	100000000000000000000000000000000000000	î	247		П		
25 EM (fin del medio) 57 9 89 Y 121 y 153 0 185 217	100000000000000000000000000000000000000		and the contract of the second			88		and Johnson				Ÿ			100000000000000000000000000000000000000	Ï	- Nicklinson		ı		
26 SUB (sustitución) 27 ESC (escape) 28 FS (sep. archivos) 29 GS (sep. grupos) 30 RS (sep. registros) 31 US (sep. unidades) 58 : 90 Z 122 Z 154 Ü 186	11000000		A CONTRACTOR OF THE PARTY OF TH	0.000		1977/27	1977	A-15550				Ö		넴	100000000000000000000000000000000000000		98,700,000	0.75	П		
27 ESC (escape) 59 ; 91 [123 { 155 ø 187 ¬ 219 ■ 251 ¹ 28 FS (sep. archivos) 60 < 92 \ 124 156 £ 188 ■ 220 ■ 252 ³ 252 3 29 GS (sep. grupos) 61 = 93] 125 } 157 Ø 189 ¢ 221 ¦ 253 ² 30 RS (sep. registros) 62 > 94 ^ 126 ~ 158 × 190 ¥ 222 Ì 254 ■ 255 nbsp	and the second second		A STATE OF THE PARTY OF THE PAR									Ü			and the second of the second	-		*:	П		
28 FS (sep. archivos) 60 < 92 \ 124 156 £ 188 20 252 3 29 GS (sep. grupos) 61 = 93] 125 } 30 RS (sep. registros) 62 > 94 ^ 126 ~ 158 × 190 ¥ 222 i 254 3 31 US (sep. unidades) 63 ? 95 _ 159 f 191 7 223 255 nbsp	(9)(-2)		11.00	100000		1000	1	The state of the s	{			65	100000000000000000000000000000000000000	=	10000000000		0.0000000000000000000000000000000000000	1	П		
29 GS (sep. grupos) 61 = 93] 125 } 157 Ø 189 ¢ 221 ; 253 ² 30 RS (sep. registros) 62 > 94 ^ 126 ~ 158 × 190 ¥ 222 i 254 ■ 31 US (sep. unidades) 63 ? 95 _ 159 f 191 ¬ 223 ■ 255 nbsp	and the same of th						i	and the second second	ì	ш			and the second second		and the last of the last of the last		THE RESERVE AND ADDRESS OF THE PERSON NAMED IN COLUMN ASSESSMENT OF THE PERSON	3	П		
30 RS (sep. registros) 62 > 94 ^ 126 ~ 158 × 190 ¥ 222 Î 254 ■ 31 US (sep. unidades) 63 ? 95 _ 159 f 191 ¬ 223 ■ 255 nbsp				150055			1	120000000	1	ш						-	100000000000000000000000000000000000000	2	П		
31 US (sep. unidades) 63 ? 95 _ 159 f 191 7 223 255 nbsp			The state of the s	1000			٨		7			1000			and the second second						
51 05 (sep. anidades) 05 1 55 _ 155 j 151 j 225 indep	100000			100,000				120	- 13				100179-00				T100 F100 F00				
177 17F1 (8)(NFIMIC)	127	DEL	(sep. unidades)	0.3		33	-				133	J	131	1	223		200	iluəp			

		eres ASCII control				res A			ASCII extendido (Página de código 437)										
00	NULL	(carácter nulo)	32			@	96		12	8	C	160	á	192	L	224	Ó		
01	SOH	(inicio encabezado)	33	!	65	A	97	a			Ÿ		-5.07.5						
02		(inicio texto)	34		66	В	98	b			1	Carac	ctere	s alta	bétic	cos m	iayúso	cula	
03		(fin de texto)	35	#	67	C	00												
04		(fin transmisión)	36	\$	68	D	100	u											
05		(consulta)	37	%	69	E	101	е	•		Desd	le el d	carác	ter 'A	'al c	aráct	er 'Z'	orden	ados
06		(reconocimiento)	38	&	70	F	102	f			on fo	rma	croci	onto					
07	BEL	(timbre)	39	•	71	G	103	g											
08	BS	(retroceso)	40	(72	Н	104	h			No se	e incl	uve a	a la 'Ñ	l' ni a	a las v	vocale	.s	
09	HT	(tab horizontal)	41)	73	1	105	i											
10	LF	(nueva línea)	42	*	74	J	106	j			acen	tuada	as (e:	stan e	en la	tabia	exter	ndida)	
11	VT	(tab vertical)	43	+	75	K	107	k	13	39	ï	171	1/2	203	TE	235	Ù		
12	FF	(nueva página)	44		76	L	108	- 1	14	10	î	172	1/4	204	ŀ	236	ý Ý		
13	CR	(retorno de carro)	45	7. 1 2	77	M	109	m	14	11	ì	173	i	205	=	237	Ý		
14	SO	(desplaza afuera)	46		78	N	110	n	14	12	Ä	174	44	206	#	238	: 37		
15	SI	(desplaza adentro)	47	1	79	0	111	0	14	13	A	175	>>	207	Ħ	239	**		
16	DLE	(esc.vínculo datos)	48	0	80	P	112	p	14	14	É	176	200	208	ð	240	=		
17	DC1	(control disp. 1)	49	1	81	Q	113	q	14	15	æ	177		209	Ð	241	±		
18	DC2	(control disp. 2)	50	2	82	R	114	r	14	16	Æ	178		210	Ê	242	_		
19	DC3	(control disp. 3)	51	3	83	S	115	S	14	17	ô	179		211	Ë	243	3/4		
20	DC4	(control disp. 4)	52	4	84	Т	116	t	14	18	Ö	180	+	212	È	244	1		
21	NAK	(conf. negativa)	53	5	85	U	117	u	14	19	ò	181	A	213	1	245	9		
22		(inactividad sínc)	54	6	86	V	118	٧	15		û	182	Â	214	1	246	÷		
23		(fin bloque trans)	55	7	87	W	119	W	15		ù	183	Α	215	1	247	3		
24		(cancelar)	56	8	88	Х	120	Х	15		ÿ	184	©	216	1	248	0		
25		(fin del medio)	57	9	89	Υ	121	у	15		Ö	185	4	217	J	249	477		
26		(sustitución)	58	:	90	Z	122	Z	15		Ü	186		218	Е	250			
27		(escape)	59	į	91	[123	{	15		Ø	187	٦	219		251	1		
28		(sep. archivos)	60	<	92	1	124	1	15		£	188	1	220		252	3		
29		(sep. grupos)	61	=	93]	125	}	15		Ø	189	¢	221	-	253	2		
30		(sep. registros)	62	>	94	٨	126	~	15		×	190	¥	222	1	254	•		
31		(sep. unidades)	63	?	95				15	9	f	191	٦	223	-	255	nbsp		
12	7 DEL	(suprimir)																	

(eres ASCII control				res A			ASCII extendido (Página de código 437)									
00	NULL	(carácter nulo)	32	espacio	64	@	96		1	128	С	160	á	192	L	224	Ó	
01	SOH	(inicio encabezado)	33		65	A	97	a	1			_			1 /			
02	STX	(inicio texto)	34		66	В	98	b				Cara	ctere	s aita	betic	cos m	ninúsc	ula
03	ETX	(fin de texto)	35	#	67	C	99	С										
04	EOT	(fin transmisión)	36	\$	68	D	100	d										
05	ENQ	(consulta)	37	%	69	E	101	е	Н	•	Desc	le el d	carác	ter 'a	'al c	aráct	:er 'z' (ordenados
06	ACK	(reconocimiento)	38	&	70	F	102	f			on fo	rma	croci	onto				
07	BEL	(timbre)	39		71	G	103	g										
08	BS	(retroceso)	40	(72	Н	104	h		•	No s	e incl	uve a	a la 'ñ	' ni a	las v	vocale:	S
09	HT	(tab horizontal)	41)	73	1	105	i										
10	LF	(nueva línea)	42	*	74	J	106	j			acen	tuada	as (es	stan e	en ia i	tabia	exten	lalaa)
11	VT	(tab vertical)	43	+	75	K	107	k		139	ï	171	1/2	203	TE.	235	Ù	
12	FF	(nueva página)	44		76	L	108	- 1		140	î	172	1/4	204	T F	236	Ý	
13	CR	(retorno de carro)	45	7,53	77	M	109	m		141	ì	173	i	205	=	237	Ý	
14	SO	(desplaza afuera)	46		78	N	110	n		142	Ä	174	44	206	#	238		
15	SI	(desplaza adentro)	47	1	79	0	111	0		143	A	175	>>	207	п	239	**	
16	DLE	(esc.vínculo datos)	48	0	80	P	112	p		144	É	176	200 200 200	208	ð	240	=	
17	DC1	(control disp. 1)	49	1	81	Q	113	q		145	æ	177	200	209	Đ	241	±	
18	DC2	(control disp. 2)	50	2	82	R	114	г		146	Æ	178		210	Ê	242	_	
19	DC3	(control disp. 3)	51	3	83	S	115	S		147	ô	179	1	211	Ë	243	3/4	
20	DC4	(control disp. 4)	52	4	84	T	116	t		148	Ö	180	+	212	È	244	1	
21	NAK	(conf. negativa)	53	5	85	U	117	u		149	ò	181	Á	213	1	245	§	
22	SYN	(inactividad sínc)	54	6	86	V	118	V		150	û	182	Â	214	ĺ	246	÷	
23	ETB	(fin bloque trans)	55	7	87	W	119	W		151	ù	183	À	215	Î	247	3	
24	CAN	(cancelar)	56	8	88	X	120	х		152	ÿ	184	©	216	Ï	248	0	
25	EM	(fin del medio)	57	9	89	Y	121	У		153	Ö	185	4	217	1	249	0.3%	
26	SUB	(sustitución)	58	:	90	Z	122	Z		154	Ü	186		218	Е	250	. • (
27	ESC	(escape)	59	;	91	[123	{		155	Ø	187	٦	219		251	1	
28	FS	(sep. archivos)	60	<	92	1	124	1		156	£	188	1	220		252	3	
29	GS	(sep. grupos)	61	=	93]	125	}		157	Ø	189	¢	221	1	253	2	
30	RS	(sep. registros)	62		94	٨	126	~		158	×	190	¥	222	1	254		
31	US	(sep. unidades)	63	?	95					159	f	191	٦	223	-	255	nbsp	
127	DEL	(suprimir)								-								

Expresiones con distintos tipos

Conversión de tipos de datos: Implícita y Explicita

Implicita:

La conversión de tipo implícita se da cuando se mezclan operandos de tipos distintos en una misma expresión. Esta conversión se realiza aplicando las siguientes reglas en el orden en que están escritas:

- si un operando es tipo 'long double', el otro se convierte a ese tipo.
- si un operando es tipo 'double', el otro se convierte a ese tipo.
- si un operando es tipo 'float', el otro se convierte a ese tipo.
- si un operando es tipo 'char' ó 'unsigned char', se convierte a 'int'.
- si un operando es tipo 'unsigned long', el otro se convierte a ese tipo.
- si un operando es tipo 'long', el otro se convierte a ese tipo.
- si un operando es tipo 'unsigned int', el otro se convierte a ese tipo.
- en otro caso, ambos son 'int'.

Explicita:

Indicando el tipo entre paréntesis.

Ejemplo:

int num = (int) (12.25/3);

Expresiones con distintos tipos

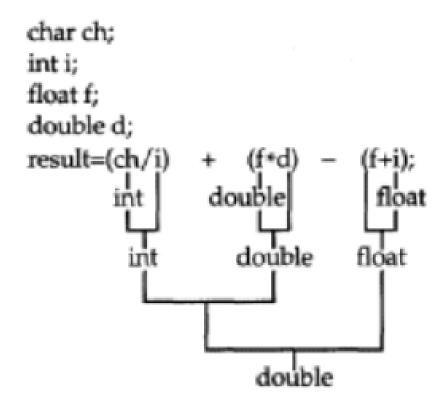


Figure 2-2 A type conversion example

```
#include <stdio.h>
int main()
    /* Declaración de variables */
    int Npizzas;
    float harina, sal, aceite, levadura, azucar, agua;
    /* Algoritmo */
    printf("Ingrese la cantidad de pizzas: ");
    scanf("%d", &Npizzas);
    harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

```
azucar = Npizzas/5.0;
agua = Npizzas/5.0 * 700;
printf("Para amasar %d pizzas necesita:\n", Npizzas);
printf("%f Kg de harina 000\n", harina);
  Impresión de una cadena constante
prin
            aradas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

```
#include <stdio.h>
int main()
    /* Declaración de variables */
                Impresión de variables mezclada con
    int Npiz
                                 texto
    float ha
    /* Algoritmo */
    printf("Ingrese la cantidad de pizzas: ");
    scanf("%d", &Npizzas);
    harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

```
azucar = Npizzas/5.0;
agua = Npizzas/5.0 * 700;
printf("Para amasar %d pizzas necesita:\n", Npizzas);
printf("%f Kg de harina 000\n", harina);
printf("%f cucharadas de sal\n", sal);
printf("%f cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

```
#include <stdio.h>
                                                              azucar = Npizzas/5.0;
                                                              agua = Npizzas/5.0 * 700;
int main()
                                                               print*("Para amasar %d pizzas necesita:\n",
    /* Decla
                                                                                                          Npizzas);
                         Cadena de control
                                                                 htf("%f Kg de harina 000\n", harina);
   int Npiz
                Contiene el texto a mostrar en pantalla
    float ha
                                                               printf("%f cucharadas de sal\n", sal);
                Donde se quiere mostrar el valor de
                                                              printf("%f cucharadas de aceite\n", aceite);
                una variable se intercalan
                                                               printf("%f gramos de levadura\n", levadura);
    /* Algor
                ESPECIFICADORES DE FORMATO
                                                               printf("%f cucharadas de azucar\n", azucar);
   printf("
                El especificador de formato depende
                                                              printf("%f cc de agua\n", agua);
                del tipo de la variable que se quiere
   scanf("%
                mostrar y del formato
                                                              return 0;
   harina =
   sal = Np
    aceite = Npizzas/3.0
    levadura = Npizzas/5.0 * 50;
```

Especificadores de formato

- Comienza con %
- Finaliza con un carácter "indicador de tipo" que depende del tipo del valor que se quiere mostrar y del formato
- Los más usados son:
 - %d: enteros
 - %f : reales (float)
 - %lf: reales (double)
 - %c: caracteres
 - %s: cadenas
- Poniendo un número entre el % y el indicador de tipo se reserva un "Ancho de campo" mínimo para la impresión y el valor queda alineado a derecha:
 - %5d
 - %20s
- Un punto y un número luego del ancho de campo indica la "precisión" con la que se va a mostrar un real
 - %5.2f
 - %.2lf

ables

```
azucar = Npizzas/5.0;
 gua = Npizzas/5.0 * 700;
       "Para amasar |%d | pizzas necesita:\n", Npizzas);
printf("%f Kg de harina 000\n", harina);
printf("%f cucharadas de sal\n", sal);
printf(|%f| cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf(|%f| cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

```
aceite = Npizzas/5.0 * 3;
levadura = Npizzas/5.0 * 50;
```

El modificador *type* del especificador de formato puede ser:

Código	Formato
% C	Carácter
% d	Enteros con signo
% i	Enteros con signo
% Id	Enteros con signo <i>long</i>
% е	Coma flotante, notación científica (e minúscula)
% E	Coma flotante, notación científica (E mayúscula)
% f	Coma flotante <i>float</i>
% If	Coma flotante double
% Lf	Coma flotante <i>long double</i>
% g	Usa e o f, el más corto
% G	Usa E o f, el más corto
% 0	Octal sin signo
% s	Cadena de caracteres
% u	Enteros decimales sin signo
% X	Hexadecimales sin signo (letras minúsculas:a - f)
% X	Hexadecimales sin signo (letras mayúsculas:A –F)
% p	Mostrar un puntero
% n	El argumento asociado será un puntero a entero al
	que se le asigna el número de caracteres escritos
% %%	Visualizar el signo %

Lista de parámetros

- Contienen variables o expresiones cuyo valor se quiere mostrar (separados por coma) |
- Debe haber un modificador de formato para cada parámetro de la lista

```
int Npizzas;
float harina, sal, aceite, levadura, azucar, agua;
/* Algoritmo */
printf("Ingrese la cantidad de pizzas: ");
scanf("%d", &Npizzas);
harina = Npizzas/5.0;
sal = Npizzas/5.0 * 2;
aceite = Npizzas/5.0 * 3;
levadura = Npizzas/5.0 * 50;
```

ables

```
azucar = Npizzas/5.0;
Npizzas/5.0 * 700;
```

```
printf("Para amasar %d pizzas necesita:\n", Npizzas);
printf("%f Kg de harina 000\n", harina);
printf("%f cucharadas de sal\n", sal);
printf("%f cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

Ejemplo básico con va

```
int main()
    /* Declaración de variables */
   int Npizzas;
   float harina, sal, aceite, levadura, azucar, agua;
    /* Algoritmo */
    printf("Ingrese la cantidad de pizzas: ");
    scanf("%d", &Npizzas);
   harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

#include <stdio.h>

Lectura de datos por teclado

- Una función muy usada es scanf() que al igual que printf() requiere la inclusión de stdio.h
- Permite la lectura de distintos tipos de valores
- Si se invoca numerosas veces a scanf() conviene invocar a la función fflush(stdin); para evitar errores

```
printf("%f cucharadas de sal\n", sal);
printf("%f cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

Ejemplo básico con va

```
#include <stdio.h>
int main()
    /* Declaración de variables */
    int Npizzas;
    float harina, sal, aceite, levadura, azucar, agua
    /* Algoritmo */
    printf("Ingrese to cantidad de pizzas: ");
    scanf("%d", &Npizzas);
    harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

Cadena de control

- Incluye el o los modificadores de formato
- Son un subconjunto de los utilizados por printf()

```
%f Kg de harina 000\n", harina);
printf("%f cucharadas de sal\n", sal);
printf("%f cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```

Código	Significado
% С	Lee un único carácter
% d	Lee un entero
% i	Lee un entero
% ld	Lee un entero de tipo <i>long</i>
% e	Lee un número en coma flotante
% f	Lee un número en coma flotante
% If	Lee un número en coma flotante double
%Lf	Lee un número en coma flotante long double
% g	Lee un número en coma flotante
% 0	Lee un número octal
% S	Lee una cadena
% X	Lee un número hexadecimal
% p	Lee un puntero
% n	Recibe un valor entero igual al número de caracteres leídos
% u	Lee un entero sin signo

Ejemplo básico con va

```
#include <stdio.h>
int main()
    /* Declaración de variables */
    int Npizzas;
    float harina, sal, aceite, levadura, azucar, agua;
    /* Algoritmo */
    printf("Ingrese la cantidad de pizzas: ");
    scanf("%d", &Npizzas);
    harina = Npizzas/5.0;
    sal = Npizzas/5.0 * 2;
    aceite = Npizzas/5.0 * 3;
    levadura = Npizzas/5.0 * 50;
```

Lista de parámetros

- Lista de variables donde se almacenarán los valores ingresados (separadas por coma)
- Deben ser precedidas por el operador &

```
kg de harina 000\n", harina);
printf("%f cucharadas de sal\n", sal);
printf("%f cucharadas de aceite\n", aceite);
printf("%f gramos de levadura\n", levadura);
printf("%f cucharadas de azucar\n", azucar);
printf("%f cc de agua\n", agua);
return 0;
```